Разбор олимпиады RuCTF 2017

3ch

В данном задании участникам предлагался для анализа сайт, на котором можно оставлять картинки и смотреть картинки от других пользователей. В интерфейсе сайта, кроме формы аутентификации есть также форма для смены пароля. Важным замечанием является то, что все формы на сайте посылаются через метод GET и у форм нет CSRF токена.

Если попытаться отправить произвольный текст вместо ссылки на картинку, можно понять, что фильтрации нет и любой текст принимается и затем будет выдан в качестве ссылки на картинку.

Интересной особенностью браузеров является то, что они загружают множество вещей, помимо самого сайта: картинки, скрипты и т.д. Из того, что периодически в выдаче появляются новые картинки, можно сделать вывод, что что-то открывает на другой стороне браузер и заполняет форму добавления картинки.

Таким образом мы можем встроить CSRF атаку, меняющую пароль, в адрес картинки. В следующий раз когда проверяющая система будет загружать страницу, она загрузит в том числе и вредоносную ссылку, которая поменяет пароль и позволит войти за того, кто добавляет картинки. Флаг демонстрируется при заходе из под этого аккаунта.

Выборы капитана

Данная задача не требовала никаких специфических знаний, кроме основ теории вероятностей. В терминах MixNet'a, пусть у нас присутствует m миксеров, n голосов (важно, что n на 1 меньше, чем число голосующих в задаче, потому что потенциальный злоумышленник один из них), k из которых проверяются при каждой передаче. Нам необходимо определить, какое может быть максимальное k такое, чтобы вероятность, получив весь трафик, найти хотя бы одно соответствие между голосующим и его голосом.

Пусть P(v_i) - вероятность того, что удастся оттрейсить голосователя i. Заметим, что в силу того, что один голосующий ничем не отличается от другого, все P(v_i) равны между собой, обозначим эту вероятность p. Тогда вероятность того, что хотя бы один наблюдатель окажется скомпрометированным, будет

1 - (1 - p)^n

Посчитаем теперь вероятность p. Для этого заметим, что чтобы можно было успешно отследить данного пользователя, он должен попасть в выбранные для проверки сообщения на всех миксерах, т.е.

$p = (k / n)^m$

Итого, если это все меньше, чем 0.01, то

$1 - (1 - (k / n)^m)^n < 0.01$
$k < n * (1 - (1 - 0.01)^(1 / n))^(1 / m)$

Флаг

Участнику дан png-файл с однотонным изображением. В файле есть 3 чанка: IHDR, IDAT и IEND. Если открыть файл в текстовом редакторе, видно, что IHDR и IEND настолько коротки, что не могут содержать в себе флага, к тому же IHDR содержит в основном нулевые байты. Подозрение падает на IDAT, т.к. он слишком длинный для однотонной картинки. IDAT сжат алгоритмом Deflate. IDAT содержит информацию о пикселях изображения построчно.

Перед каждой строкой пикселей стоит байт, содержащий настройку фильтрации этой строки пикселей. Фильтрация подготавливает строку пикселей для лучшего сжатия. Байт настройки фильтрации имеет 5 допустимых значений - существует 5 фильтров. Изображение полностью однотонное, все строки пикселей одинаковы, но байт фильтрации меняется от строки к строке. Значит, флаг спрятан там.

Все 5 допустимых значения байта используются в изображении, поэтому естественно будет предположить, что информация закодирована в пятеричной системе счисления. Составляем из байтов фильтрации 5-ричное число, переводим его в 256-ричное число и получаем последовательность байт, кодирующую текст, яляющийся ответом на задачу.

Аттракцион

По намекам в условии можно сопоставить даты и найти язык TRAC, который был разработан в 1959 году, а реализован в 1964. Далее достаточно найти интерпретатор языка Truc и запустить программу, данную в условии, вывод программы и будет ответом.

Проблема могла возникнуть при использовании более новых или не работающих версий интерпретатора TRAC, но в силу небольшого размера набора команд, это не было большим препятствием.